IN THE CLAIMS

1. (Currently Amended) A method of modeling, building and implementing a software application on a remote deployment system corresponding to a base application comprising:

identifying a set of objects in the base application for inclusion in the remote deployment and operable by an alternate control path;

translating, via an object translator, the identified set of objects into a set of remote application objects parallel to the objects in the base application, the identified set of objects defining a graphical user interface operable to interact with a user;

deploying the translated remote application objects on a remote server; and

generating, from at least a subset of the translated remote application objects, executable objects executable by a server runtime engine at the remote server, the server runtime engine operable to generate transportable objects corresponding to the generated executable objects, the transportable objects further operable to generate, via the alternate control path, GUI executable objects on a remote client runtime engine, the remote client runtime engine responsive to the transportable objects to generate the corresponding GUI executable objects, the set of objects in the base application further comprising GUI objects and processing objects, translating further comprising:

determining, via an object classifier in the object translator, if the object is a GUI object or a processing object; and

if the object is a GUI object, generating a reference to the server runtime engine.

2. (Original) The method of claim 1 wherein translating comprises generating, via a label mapper in the object translator, a corresponding remote application object for each identified object in the base application, the generated remote application object operable for execution in the remote deployment.

3. (Canceled)

4. (Original) The method of claim 1 further comprising identifying, from the set of objects in the base application, GUI objects and processing objects, the GUI objects responsible for producing GUI display elements including at least one of GUI screens, GUI icons, GUI controls, GUI buttons and GUI selections.

5. (Original) The method of claim 4 wherein translating further comprises:

identifying, via an association manager in the object translator, associations between the remote application objects and the GUI display elements; and

storing, in an associated object table, the identified associations.

6. (Original) The method of claim 5 wherein the associations are further indicative of associations between the remote application objects and external references in a data source.

7. (Original) The method of claim 1 wherein the base application further includes a data source interface to an external data source, and translating further comprises identifying object references from the translated remote application objects into the data source, the data source responsive to the corresponding object reference.

8. (Currently Amended) The method of claim 7 wherein the base application is a SAN management application, the data source is a SAN management server and the external references are indicative of a manageable entity in the SAN, the SAN management server operable to store and retrieve information about the manageable entities in the SAN.

9. (Original) The method of claim 1 wherein the remote deployment comprises a web server and a browser, the web server operable to include the translated objects and the server runtime engine and the browser adapted to include the client runtime engine in communication with the server runtime engine via the alternate control path, the alternate control path comprising an API portal, the API portal including an Internet connection.

10. (Original) The method of claim 1 wherein translating further comprises:

determining, for each of the translated application objects, overloaded methods corresponding to GUI display elements; and

resolving, for the GUI display elements, style inconsistencies in the GUI display produced by the client runtime engine.

11. (Original) The method of claim 1 wherein translating further comprises:

determining base application objects employing compound GUI display elements;

computing an aggregation of unary display elements consistent with the determined compound GUI display elements; and

modifying the translated application object such that the client runtime engine employs the aggregated unary display elements.

12. (Original) The method of claim 7 wherein the base application is a storage area network (SAN) management application and the data source is a SAN management server having a database of manageable entities (ME) for providing storage data services via the SAN, each of the manageable entities responsive to the SAN management application and further wherein the GUI elements represent SAN elements, the SAN elements corresponding to the manageable entities in the SAN and the transportable objects for reporting status

of the manageable entities from corresponding agent components in the SAN, each of the agent components corresponding to at least one manageable entity, the agent components further responsive to the return transportable objects for managing the manageable entities in the SAN.

13. (Original) The method of claim 5 wherein the associations are further indicative of relationships between GUI elements and the executable objects in the server runtime engine.

14. (Original) The method of claim 6 wherein the associations are further indicative of references between manageable entities indicated in the SAN management server database.

15. (Currently Amended) A computing device having a memory, processor, and an interface for modeling, building and implementing a software application on a remote deployment system corresponding to a base application comprising:

an interface to a set of base application objects operable to identify and access a set of objects in the base application for inclusion in the remote deployment and operable by an alternate control path;

an object translator operable to translate the identified set of objects into a set of remote application objects parallel to the objects in the base application, the identified set of objects defining a graphical user interface operable to interact with a user, the translated set of identified objects operable for deployment on a remote server, the remote server further operable to generate, from at least a subset of the translated remote application objects, executable objects executable by a server runtime engine at the remote server, the server runtime engine operable to generate transportable objects corresponding to the generated executable objects, the transportable objects further operable to generate, via the alternate control path, GUI executable objects on a remote

client runtime engine, the remote client runtime engine responsive to the transportable objects to generate the corresponding GUI executable objects, the object translator further comprising:

a label mapper operable to generate a corresponding remote application object for each identified object in the base application, the generated remote application object operable for execution in the remote deployment; and

an object classifier, wherein the set of objects in the base application further comprises GUI objects and processing objects, the object translator operable to determine, if the object is a GUI object or a processing object.

16. (Canceled)

17. (Original) The computing device of claim 15 further comprising an object classifier in the object translator, wherein the set of objects in the base application further comprises GUI objects and processing objects, and wherein the object translator is operable to:

determine, if the object is a GUI object or a processing object; and

if the object is a GUI object, operable to generate a reference to the server runtime engine.

18. (Original) The computing device of claim 15 wherein the object classifier is further operable to identify, from the set of objects in the base application, GUI objects and processing objects, the GUI objects responsible for producing GUI display elements including at least one of GUI screens, GUI icons, GUI controls, GUI buttons and GUI selections.

19. (Original) The computing device of claim 17 wherein the object translator further includes an association manager operable to:

identify, via an association manager in the object translator, associations between the remote application objects and the GUI display elements; and

store, in an associated object table, the identified associations.

20. (Original) The computing device of claim 19 wherein the associations are further indicative of associations between the remote application objects and external references in a data source.

21. (Original) The computing device of claim 15 wherein the base application further includes a data source interface to an external data source, and translating further comprises identifying object references from the translated remote application objects into the data source, the data source responsive to the corresponding object reference.

22. (Currently Amended) The computing device of claim 20 wherein the base application is a SAN management application, the data source is a SAN management server and the external references are indicative of a manageable entity in the SAN, the SAN management server operable to store and retrieve information about the manageable entities in the SAN.

23. (Original) The computing device of claim 15 wherein the remote deployment comprises a web server and a browser, the web server operable to include the translated objects and the server runtime engine and the browser adapted to include the client runtime engine in communication with the server runtime engine via the alternate control path, the alternate control path comprising an API portal, the API portal including an Internet connection.

24. (Original) The computing device of claim 15 wherein the object classifier further includes an overload parser operable to:

determine, via the overload parser in the object classifier, for each of the translated application objects, overloaded methods corresponding to GUI display elements; and

resolve, for the GUI display elements, style inconsistencies in the GUI display produced by the client runtime engine.

25. (Original) The computing device of claim 15 wherein the association manger further includes a display element validator operable to:

determine base application objects employing compound GUI display elements;

compute an aggregation of unary display elements consistent with the determined compound GUI display elements; and

modify, via the display element validator, the translated application object such that the client runtime engine employs the aggregated unary display elements.

26. (Original) The computing device of claim 21 wherein the base application is a storage area network (SAN) management application and the data source is a SAN management server having a database of manageable entities (ME) for providing storage data services via the SAN, each of the manageable entities responsive to the SAN management application and further wherein the GUI elements represent SAN elements, the SAN elements corresponding to the manageable entities in the SAN and the transportable objects for reporting status of the manageable entities from corresponding agent components in the SAN, each of the agent components corresponding to at least one manageable entity, the agent components further responsive to the return transportable objects for managing the manageable entities in the SAN.

27. (Original) The computing device of claim 19 wherein the associations are further indicative of relationships between GUI elements and the executable objects in the server runtime engine.

28. (Original) The computing device of claim 20 wherein the associations are further indicative of references between manageable entities indicated in the SAN management server database.

29. (Currently Amended) The method of claim 12 wherein the SAN management application is operable to correlate a plurality of manageable entities with a plurality of agent components, each of the manageable entities and agent components having a corresponding executable object in the set of objects in base application, wherein translating further comprises:

identifying objects in the base application corresponding to the operator console;

identifying objects in the base application corresponding to the data source, the data source further comprising a control centern ~~E~~(CC) database operable to store information corresponding to each of the manageable entities according to a predetermined hierarchy defined in the base application;

copying the identified objects in the base application corresponding to the operator console to the remote web server;

determining, in the copied objects, references for operator input and feedback via the console at the base application;

redirecting the determined references to the remote web client, the remote web client in communication with the remote web server by a public access mechanism;

examining the copied objects in the remote web server to identify overloaded methods and compound display elements;

modifying, according to design porting logic, the identified objects having overloaded methods and compound display elements for operation on the remote web server; and

deploying the copied objects as executable objects on the server runtime engine in the web server.

30. (Currently Amended) A computer program product having a computer readable storage medium operable to store computer program logic embodied in an encoded set of processor based instructions defined as computer program code encoded thereon and executable by a processor responsive to the instructions for modeling, building and implementing a software application on a remote deployment system corresponding to a base application comprising:

computer program code for identifying a set of objects in the base application for inclusion in the remote deployment and operable by an alternate control path;

identifying, from the set of objects in the base application, GUI objects and processing objects, the GUI objects responsible for producing GUI display elements;

computer program code for translating, via an object translator, the identified set of objects into a set of remote application objects parallel to the objects in the base application, the identified set of objects defining a graphical user interface operable to interact with a user;

computer program code for deploying the translated remote application objects on a remote server; and

computer program code for generating, from at least a subset of the translated remote application objects, executable objects executable by a server runtime engine at the remote server, the server runtime engine operable to generate transportable objects corresponding to the generated executable objects, the transportable objects further operable to generate, via the alternate

control path, GUI executable objects on a remote client runtime engine, the remote client runtime engine responsive to the transportable objects to generate the corresponding GUI executable objects, translating further comprising:

identifying, via an association manager in the object translator, associations between the remote application objects and the GUI display elements; and

storing, in an associated object table, the identified associations.

31. (Currently Amended) An encoded set or processor based instructions on a computer readable storage medium defined as program code executable by a processor responsive to the instructions ~~computer data signal~~ for modeling, building and implementing a software application on a remote deployment system corresponding to a base application comprising:

program code for identifying a set of objects in the base application for inclusion in the remote deployment and operable by an alternate control path;

program code for translating, via an object translator, the identified set of objects into a set of remote application objects parallel to the objects in the base application, the identified set of objects defining a graphical user interface operable to interact with a user;

program code for deploying the translated remote application objects on a remote server; and

program code for generating, from at least a subset of the translated remote application objects, executable objects executable by a server runtime engine at the remote server, the server runtime engine operable to generate transportable objects corresponding to the generated executable objects, the transportable objects further operable to generate, via the alternate control path, GUI executable objects on a remote client runtime engine, the remote client runtime engine responsive to the transportable objects to generate the corresponding GUI executable objects.

32. (Currently Amended) A computing device having a memory, processor, and an interface for modeling, building and implementing a software application on a remote deployment system corresponding to a base application comprising:

means for identifying a set of objects in the base application for inclusion in the remote deployment and operable by an alternate control path;

means for translating, via an object translator, the identified set of objects into a set of remote application objects parallel to the objects in the base application, the identified set of objects defining a graphical user interface operable to interact with a user, the means for translating comprising means for generating, via a label mapper in the object translator, a corresponding remote application object for each identified object in the base application, the generated remote application object operable for execution in the remote deployment;

means for deploying the translated remote application objects on a remote server; and

means for generating, from at least a subset of the translated remote application objects, executable objects executable by a server runtime engine at the remote server, the server runtime engine operable to generate transportable objects corresponding to the generated executable objects, the transportable objects further operable to generate, via the alternate control path, GUI executable objects on a remote client runtime engine, the remote client runtime engine responsive to the transportable objects to generate the corresponding GUI executable objects, the set of objects in the base application further comprising GUI objects and processing objects, the means for translating further comprising:

means for determining, via an object classifier in the object translator, if the object is a GUI object or a processing object; and

means for generating, if the object is a GUI object, a reference to the server runtime engine,

means for identifying, via an association manager in the object translator, associations between the remote application objects and the GUI display elements; and

means for storing, in an associated object table, the identified associations,

the base application further including a data source interface to an external data source, the means for translating further comprising:

means for identifying object references from the translated remote application objects into the data source, the data source responsive to the corresponding object reference, wherein the base application is a SAN management application, the data source is a SAN management server and the external references are indicative of a manageable entity in the SAN, the SAN management server operable to store and retrieve information about the manageable entities in the SAN.